Таблица 36. Варианты задач

Вариант	Константа
1	Первой константы (начальное значение 01000000000000000000000000000000)
2	Второй константы (начальное значение 02000000000000000000000000000000)
3	Третьей константы (начальное значение 03000000000000000000000000000000)
4	Четвертой константы (начальное значение 04000000000000000000000000000000)
5	Пятой константы (начальное значение 05000000000000000000000000000000)
6	Шестой константы (начальное значение 06000000000000000000000000000000)
7	Седьмой константы (начальное значение 07000000000000000000000000000000)
8	Восьмой константы (начальное значение 08000000000000000000000000000000)
9	Девятой константы (начальное значение 09000000000000000000000000000000)
10	Десятой константы (начальное значение 0A000000000000000000000000000000)

DES

Алгоритм DES (Data Encryption Standard) является блочным симметричным алгоритмом шифрования, разработанным в 1970-х годах. Он используется для обеспечения конфиденциальности данных путем их обратимого шифрования. DES был разработан для замены более ранних алгоритмов шифрования и был принят в качестве федерального стандарта шифрования данных в США в 1977 году.

Процесс шифрования состоит из начальной перестановки, 16 циклов шифрования и конечной перестановки.

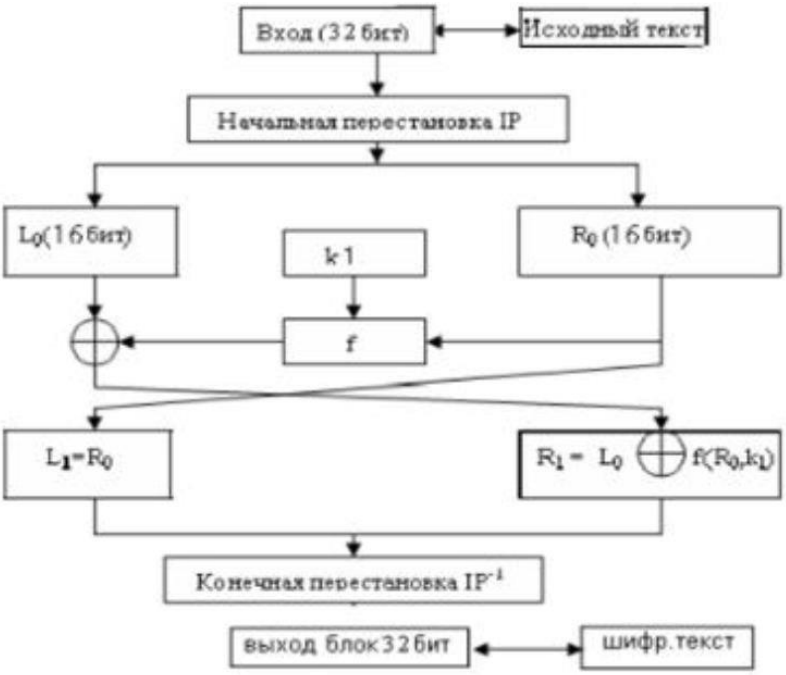

Рисунок 40. Схема DES/y

Для понимания работы алгоритма будет использоваться DES/y с упрощенными расчетами.

Начальная перестановка IP

IP							
26	18	10	2	28	20	12	4
30	22	14	6	32	24	16	8
25	17	9	1	27	19	11	3
29	21	13	5	31	23	15	7

Конечная перестановка IP^{-1}

IP^{-1}							
20	4	24	8	28	12	32	16
19	3	23	7	27	11	31	15
18	2	22	6	26	10	30	14
17	1	21	5	25	9	29	13

Рисунок 41. IP перестановки

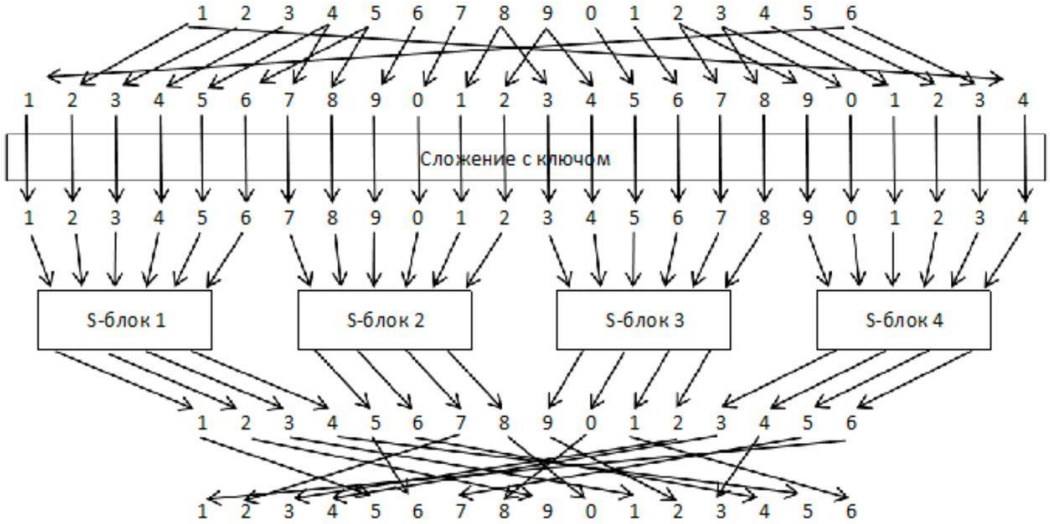

Рисунок 42. Функция расширения

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7	
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8	S_1
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0	
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13	
0	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10	
1	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5	S_2
2	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15	
3	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9	
0	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8	
1	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1	S_3
2	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7	
3	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12	
0	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15	
1	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9	S_4
2	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4	
3	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14	

Рисунок 43. S-блок для DES/y

16	7	12	13	1	5	15	10	2	8	3	9	14	6	4	11	P-box

Рисунок 44. P-box

Пример

Исходное сообщение: КТСО

Ключ: КИБ

К	11	00001011
Т	19	00010011
С	18	00010010
О	15	00001111

К	11	00001011
И	09	00001001
Б	01	00000001

Перестановка по IP

00001011
00010011
00010010
00001111

→

00000110	**L0**
10001011	
00000000	**R0**
10011111	

L1 = R0

Функция расширения f (16 бит -> 24 бит):

0000000010011111

100000000001010011111110

Сложение с ключом XOR(f, Key)

100000000001010011111110

1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	1	1	1	1	1	1	0

⊕

0	0	0	0	1	0	1	1	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1

=

1	0	0	0	1	0	1	1	0	0	0	1	1	1	0	1	1	1	1	1	1	1	1	1

Операция S-block

100010110001110111111111

Разделение на блоки по 6 символов

['100010', '110001', '110111', '111111']

Получение (строка, столбец)/(x_i, y_i)

[(2, 1), (3, 8), (3, 11), (3, 15)]

Значения (x_i, y_i) в S_i-блоке

[1, 11, 3, 14]

Перевод в бинарный код длиной 4

['0001', '1011', '0011', '1110']

Получение исходной последовательности бит

0001101100111110

Операция P-box

0001101100111110

0111011001001011

R1 = XOR(P_box, L0)

0	1	1	1	0	1	1	0	0	1	0	0	1	0	1	1

$\oplus$

0	0	0	0	0	1	1	0	1	0	0	0	1	0	1	1

=

0	1	1	1	0	0	0	0	1	1	0	0	0	0	0	0

Перестановка по $\boldsymbol{IP^{-1}}$

L1	00000000	→	10000101
	10011111		10000001
R1	01110000		10001001
	11000000		00001101

Конечный перевод

10000101	85
10000001	81
10001001	89
00001101	0D

Ответ: 8581890D

Задача: Имея исходное сообщение и ключ, получить зашифрованное сообщение.

Вариант 1

Сообщение	ТАСК
Ключ	БИТ

Вариант 2

Сообщение	ГУРУ
Ключ	МИФ

Вариант 3

Сообщение	ДЖУН
Ключ	БАГ

Вариант 4

Сообщение	МИДЛ
Ключ	АПИ

Вариант 5

Сообщение	ЮЗЕР
Ключ	ТИК

Вариант 6

Сообщение	ГАЙД
Ключ	ЛАГ

Вариант 7

Сообщение	ЛИСТ
Ключ	ДОМ

Вариант 8

Сообщение	ПАТЧ
Ключ	КОД

Вариант 9

Сообщение	СОФТ
Ключ	СМС

Вариант 10

Сообщение	ФЭЙК
Ключ	ХИТ

ПОТОЧНЫЕ ШИФРЫ

Регистры сдвига

Большинство реальных поточных шифров основано на регистрах сдвига с обратной связью. Регистр сдвига применяют для генерации ключевой последовательности.

Регистр сдвига с обратной связью состоит из двух частей: регистра сдвига и функции обратной связи. Регистр сдвига представляет собой последовательность битов. (Количество бит определяется длиной сдвигового регистра. Если длина равна n битам, то регистр называется n-битовым регистром сдвига).

Всякий раз, когда нужно извлечь бит, все биты регистра сдвигаются вправо на 1 позицию. Новый крайний левый бит, полученный от функции обратной связи, является функцией всех остальных битов регистра. На выходе регистра оказывается вытесненный младший значащий бит.

Периодом регистра называется длина получаемой последовательности до начала ее повторения.

Простейшим типом регистров сдвига является регистр сдвига с линейной обратной связью или РСЛОС (рис. 45). Двоичные псевдослучайные периодические последовательности, генерируемые с использованием регистров сдвига с линейной обратной связью (как правило, аппаратным способом), именуются РСЛОС-последовательностями или линейными рекуррентными последовательностями.

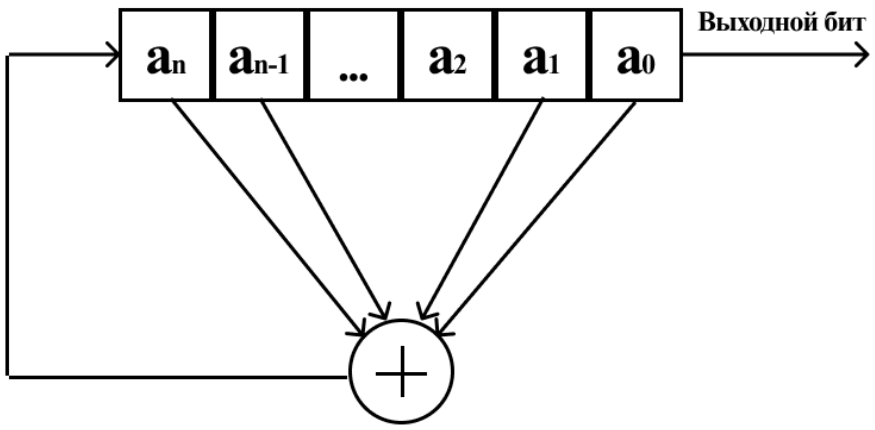

Рисунок 45. Схема работы регистра сдвига с линейной обратной связью

Формально n-битовый РСЛОС можно описать многочленом степени n от формальной переменной x, где i-му биту соответствует член x^i с коэффициентом 0, если бит входит в последовательность, или с коэффициентом 1, если не входит.

Свободный член многочлена всегда равен 1. Например, для многочлена x^5+x^2+x+1 регистр будет по схеме (рис. 46):

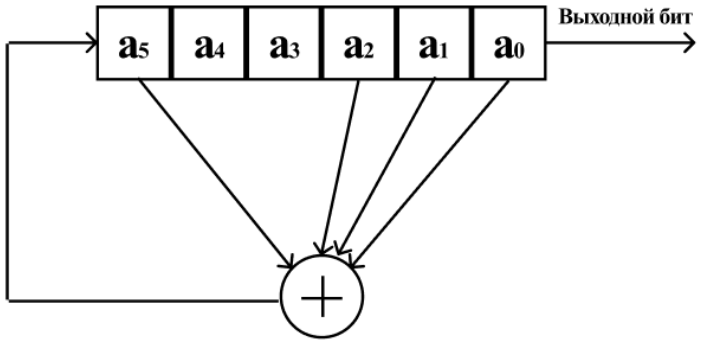

Рисунок 46. Схема РСЛОС

Пример

Регистр сдвига задан выражением x^4+1. Входная последовательность a=10110. Определить выходную последовательность b и период n.

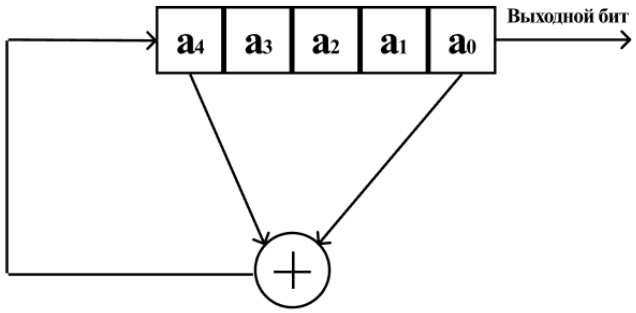

Рисунок 47. Схема РСЛОС

Таблица 37. Пример

Номер такта	Состояние регистра					Выходной бит
	a4	a3	a2	a1	a0	
Начальное значение	1	0	1	1	0	-
1	1	1	0	1	1	0
2	0	1	1	0	1	1
3	1	0	1	1	0	1

Ответ: b=011, n=3

Математические модели РСЛОС:

1) Конфигурация Фибоначчи. В конфигурации Фибоначчи в зависимости от многочлена обратной связи выбираются участвующие в сумме по модулю два ячейки памяти. I-тая ячейка участвует в сумме если в многочлене присутствует i-тая степень. Выходной бит должен складываться по модулю два с другими битами, прежде чем заполнить первый бит (рис. 48):

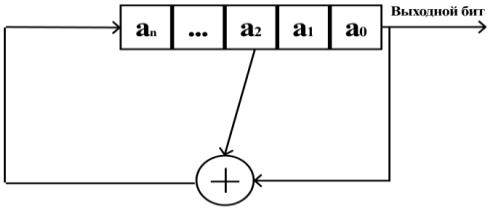

Рисунок 48. Схема РСЛОС конфигурации Фибоначчи

Пример

Многочлен обратной связи второй степени задан выражением x^2. Входная последовательность a=101. Определить выходную последовательность b и период n.

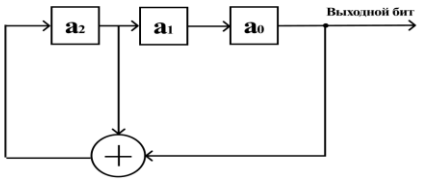

Рисунок 49. Пример регистра конфигурации Фибоначчи, заданный выражением x^2

Таблица 38. Пример

Номер такта	Состояние регистра			Выходной бит
Начальное значение	a2	a1	a0	
	1	0	1	
0	1	0	1	1
1	0	1	0	0
2	0	0	1	1
3	1	0	0	0
4	1	1	0	0
5	1	1	1	1
6	0	1	1	1
7	1	0	1	1

Ответ: b=1010011, n=7.

2) Конфигурация Галуа. В конфигурации Галуа для генерации нового состояния выполняется операция XOR с выходным битом, заменяя старый бит участвующих в операции ячеек памяти. Участие ячейки определяется аналогично наличием i-й степени в многочлене. (рис. 50)

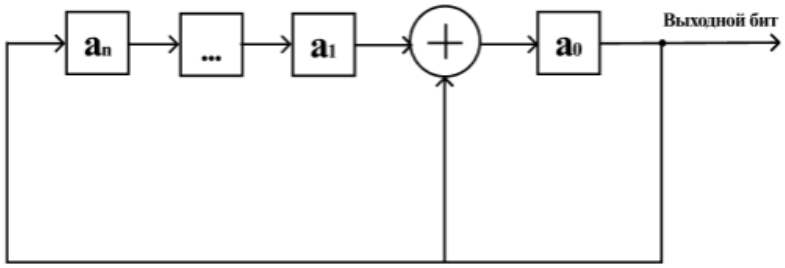

Рисунок 50. Схема РСЛОС конфигурации Галуа

Пример

Многочлен обратной связи второй степени задан выражением x^2+x. Входная последовательность a=100. Определить выходную последовательность b и период n.

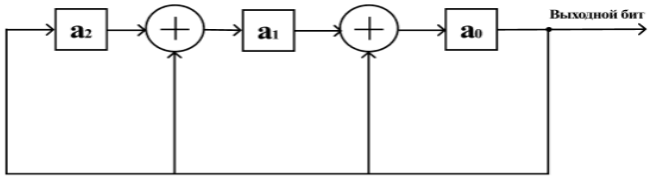

Рисунок 51. Пример регистра конфигурации Галуа, заданный выражением x^2+x

Таблица 39. Пример

Номер такта	Состояние регистра			Выходной бит
Начальное значение	a2	a1	a0	
	1	0	0	
0	1	0	0	0
1	0	1	0	0
2	0	0	1	1
3	1	1	1	1
4	1	0	0	0

Ответ: b=0011, n=4.

Задача: Определить выходную последовательность b и период n.

Вариант 1

Регистр сдвига задан выражением задан выражением x^3+x+1. Входная последовательность a=1011.

Вариант 2

Регистр сдвига (3-битовый) конфигурации Фибоначчи задан выражением задан выражением x+1. Входная последовательность a=101.

Вариант 3

Регистр сдвига конфигурации Галуа задан выражением задан выражением x^3+x. Входная последовательность a=1011.

Вариант 4

Регистр сдвига задан выражением задан выражением x^3+x+1. Входная последовательность a=1000.

Вариант 5

Регистр сдвига конфигурации Фибоначчи задан выражением задан выражением x. Входная последовательность a=110.

Вариант 6

Регистр сдвига задан выражением x^3+x^2+1. Входная последовательность a=1010.

Вариант 7

Регистр сдвига (3-битовый) конфигурации Фибоначчи задан выражением x^2. Входная последовательность a=001.

Вариант 8

Регистр сдвига конфигурации Галуа задан выражением x^3+x^2. Входная последовательность a=1100.

Вариант 9

Регистр сдвига задан выражением x^3+x^2+1. Входная последовательность a=1101.

Вариант 10

Регистр сдвига конфигурации Фибоначчи задан выражением x. Входная последовательность a=011.

A3

A3 — алгоритм, используемый в процессе аутентификации в глобальном цифровом стандарте для мобильной сотовой связи GSM. A3 является, таким образом, элементом системы обеспечения конфиденциальности разговора в GSM наряду с алгоритмами A5 и A8. Задача алгоритма — генерация отзыва (SRES — Signed Response) на случайный пароль (RAND — Random),